ARCADIAN

Volume 2 No.6

April 25,1980

BALLY GAME CARTRIDGES to be expected later this year are Demolition Derby and Dogpatch. ROM chip orders were placed at the end of March.

COMPUTER/ELECTRONIC SURPLUS FREAKS and others are invited to attend the 5th Annual California Computer Swap Meet to be held at the Santa Clara County Fairgrounds on June 1, 10 to 6. Last year's show had a mix of individuals selling flea market stuff and stores/manufacturers selling their wares. Free to buyers, $25 for flea market sellers. If interested in table space, call John Craig at 415-324-2404

NEW ADDITION to my house is a telephone line with the number 408-258-4586. (I have a teen-age daughter, nuff said)

PROJECT THREE refers to the TRS-80 interface. A few more words on this. It is planned to provide a kit with a printed circuit board and a minimum of parts installed. Four sockets would allow the insertion of 4K of RAM in 1K increments, and this will occupy the address space left vacant by the upper half of BASIC (12288 to 16383_D) so that machine code may be run.

BIBLE QUIZ is a teaching device that requires some pre-activity in loading question/answer segments. First punch in the program to line 5020, and store it on tape. Do not rewind, but let about 10 seconds of blank space be recorded. Then follow the instructions on page 49 and load a question/answer segment into the Bally. Load that onto tape, and again leave about 10 seconds of space. Continue with segments as desired. When finished, the tape can be rewound and loaded into the Bally. When the game starts, shut off the recorder and answer the question. Once that is done, start the recorder again and it will automatically replace the question/answer with a new one. If you seem to get illogical questions/answers, it means that the separation between them on the tape is not great enough, or you are moving too slowly.

SALVAGE BOARDS are here and available. I haven't had much chance to dig into them, but the are fully stuffed, with keypad switches and cartridge slot.

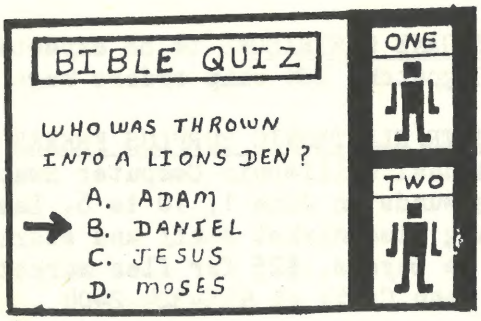
SPACE WAR CORRECTION mentioned last issue requires that line 210 be modified to read GOTO 30 instead of GOTO 130.

STATIC ELECTRICITY can cause havoc and failure of the machine. One subscriber has added a separate ground wire between the metal parts of the hand controller and the metalwork inside the Bally.

S & W DISTRIBUTING is the national distributor for all Bally products (except those sold through Montgomery Ward). Jack Nieman, formerly national sales manager for Bally Consumer Products is now actively involved in this operation, and has indicated that S & W will assist Arcade owners in purchasing hardware, with a discount for ARCADIANs. 5300B McDermott Dr. Berkeley IL 60163 312-449-5000

```
1 .BIBLE QUIZ
2 .BY B HENSEL
3 I=0;J=0
4 F=2
5 :RETURN ;CLEAR ;T=0
6 @(1)=I;@(2)=J
10 &(9)=30;BC=190;FC=40
20 &(0)=96;&(1)=96
30 &(2)=7;&(3)=7
40 CLEAR ;CY=38
50 PRINT "    BIBLE QUIZ
55 BOX -27,38,66,11,3
62 CX=55;CY=35;PRINT "ONE
70 BOX 60,20,26,40,3
72 BOX 60,30,26,1,3
74 CX=55;CY=-9;PRINT "TWO
80 BOX 60,-24,26,40,3
90 BOX 60,-14,26,1,3
95 CY=20
102 P=1;GOSUB 700
104 P=2;GOSUB 700
110 PRINT " WHO SET FIRE TO
120 PRINT " THREE HUNDRED
130 PRINT " FOXES TAILS?
135 PRINT
140 PRINT "     A. ADAM
150 PRINT "     B. RUTH
160 PRINT "     C. SAMSON
170 PRINT "     D. CAIN
180 C=3
200 P=0
210 IF TR(1)=1P=1;Y=35
220 IF TR(2)=1P=2;Y=-9
230 IF P=0GOTO 210
235 BOX 60,Y,24,8,3
240 FOR A=1TO 500;NEXT A
245 BOX 60,Y,24,8,3
250 A=1
260 CY=-5-(Ab8)
265 BOX 60,Y,24,8,3
270 PRINT "  a
280 FOR B=1TO 50
290 IF TR(P)=1GOTO 350
300 NEXT B
302 BOX 60,Y,24,8,3
305 CY=-5-(Ab8);PRINT "   "
310 A=A+1
320 IF A>4A=1;T=T+1
322 IF T<5GOTO 260
323 NT=10
324 MU=49;MU=48;MU=48;MU=48;MU=48;T=0;P=P+1;IF P=3P=1
325 Y=35;IF P=2Y=-9
329 NT=3
330 GOTO 260
350 PRINT
360 CY=-40
365 NT=10
366 Y=0
```

→49



b means multiply
c means divide

BIBLE QUIZ

By Bob Hensel

BIBLE QUIZ is a two player game. A question about the Bible is printed on the screen with four multiple choicn answers. If a player knows the answer he pulls his trigger. An arrow moves alongside the answers. When the arrow points to the proper answer the player again pulls his trigger. If he is correct a portion of a man will appear in his score box, else a portion will be taken away and the correct answer indicated The screen then goes blank. The tape recorder should then be turned on to read the next question into the computer. When the images again appear on the screen, turn off the recorder.

NOTE the new character a in line 270 This signifies the right arrow which the printer will not recognize. →

BIBLE QUIZ, cont.

```
367 IF P=2Y=-44
370 IF A=CGOSUB 500
380 IF A#CGOSUB 600
385 NT=3;FOR A=1TO 1000;NEXT A
387 IF @(P)=10GOSUB 3000
390 GOTO 5000
500 @(P)=@(P)+1
510 GOSUB 700
520 MU=49;GOSUB 595;MU=51;GOSUB 595;MU=53;GOSUB 595
530 MU=56;GOSUB 595;MU=48;GOSUB 595;MU=53;GOSUB 595
540 MU=56;GOSUB 595;MU=48;GOSUB 595;MU=48;GOSUB 595;MU=48;GOSUB 595;MU=48
550 GOSUB 595
595 BOX 60,20+Y,24,38,3
599 RETURN
600 IF @(P)=0GOTO 650
602 F=1
610 E=@(P)b10+700
620 GOSUB E
630 F=2
640 @(P)=@(P)-1
650 GOSUB 595
660 MU=34;MU=48;MU=48;MU=34;MU=48;MU=34;MU=34;MU=48;MU=48
670 MU=36;MU=48;MU=35;MU=35;MU=48;MU=34;MU=34;MU=48;MU=33;MU=34;MU=48;MU=48
675 CY=-5-(Ab8);PRINT "    ";CY=-5-(Cb8);PRINT " a
699 RETURN
700 Y=0;IF P=2Y=-44
710 IF @(P)>0BOX 60,26+Y,5,5,F
720 IF @(P)>1BOX 60,18+Y,7,10,F
730 IF @(P)>2BOX 55,18+Y,2,10,F
740 IF @(P)>3BOX 66,18+Y,2,10,F
750 IF @(P)>4BOX 58,8+Y,2,10,F
760 IF @(P)>5BOX 63,8+Y,2,10,F
770 IF @(P)>6BOX 53,14+Y,2,2,F
780 IF @(P)>7BOX 68,14+Y,2,2,F
790 IF @(P)>8BOX 56,4+Y,2,2,F
800 IF @(P)>9BOX 65,4+Y,2,2,F
830 RETURN
3000 BOX -20,-5,120,74,2
3010 CY=5;PRINT "    THE WINNER IS:
3020 PRINT
3030 PRINT "      PLAYER ",
3040 IF P=1PRINT "ONE"
3050 IF P=2PRINT "TWO"
3060 FOR A=1TO 3000;NEXT A
3070 I=0;J=0
5000 I=@(1);J=@(2)
5010 FC=BC;&(9)=50
5020 :INPUT
>
```

To record new questions for this game on a tape do the following:

```
Push RESET
5 STOP
109 .QUESTIONS
110 PRINT "WHO DISCOVERED
120 PRINT "AMERICA IN 1492?
130 PRINT "
140 PRINT "       A. WASHINGTON
150 PRINT "       B. COLUMBUS
160 PRINT "       C. CARTER
170 PRINT "       D. DESOTO
180 C=2
:PRINT;NT 1;LIST 109
```

turn on the tape recorder, press GO after the listing is done enter GOTO 5 press GO and turn off the recorder.

Note: C=1 if the answer is A,
      C=2 if the answer is B, etc.

The program can be easily changed to be used for any educational type testing game by changing the title and making appropriate questions.

Sep. 73

```
  1 .
  2 .
  3 .BLACK HOLE
 10 CLEAR ;NT=0;&(22)=200;FC=7;BC=8
 20 CY=0;PRINT "   THE BLACK HOLE"
 30 FOR A=255TO 0STEP -1;&(18)=A;NEXT A
 40 CLEAR ;CY=40
 50 PRINT "  SELECT SKILL LEVEL";PRINT "   TO ORBIT";PRINT
 60 PRINT "  1.EASY";PRINT "  2.MODRATE";PRINT "  3.HARD"
 70 INPUT "ENTER NUMBER TO GO"S
 80 H=0;O=0
 90 IF (S<1)+(S>3)GOTO 10
100 CLEAR ;FC=7;BC=0;&(22)=200;&(16)=235;&(18)=230;&(20)=200
110 FOR A=1TO 50;BOX RND (160)-80,RND (80)-40,1,1,1;NEXT A
170 X=-75;Y=40;B=0;C=0;M=0;N=15;D=-5;E=0
180 F=100;&(23)=255;BOX 0,0,7,7,1;BOX 0,0,5,5,2
200 IF M>0D=D-1
201 IF M<0D=D+1
202 IF N>0E=E-1
203 IF N<0E=E+1
210 BOX X,Y,3,3,3;BOX M,N,8,2,3;BOX M+2,N+1,1,2,3
220 I=JX(1);J=JY(1)
230 IF F<1PRINT "NO FUEL";I=0;J=0
235 IF I#0&(21)=255
236 IF J#0&(21)=255
237 IF I=0IF J=0&(21)=0
240 IF I#0BOX X-Ib2,Y,3,1,3;F=F-1
241 IF J#0BOX X,Y-Jb2,1,3,3;F=F-1
245 U=X;V=Y;B=B+I;C=C+J
250 CY=44;PRINT #4,F,#4,H,#3,O
260 IF X>0B=B-1
261 IF X<0B=B+1
262 IF Y>0C=C-1
263 IF Y<0C=C+1
265 IF X>78X=-78
266 IF X<-78X=78
267 IF Y>40Y=-40
268 IF Y<-40Y=40
270 X=X+B;Y=Y+C
280 IF X<5IF X>-5IF Y<5IF Y>-5GOTO 1000
290 IF S=1IF X>14IF X<16IF Y<10IF Y>-10IF C=5GOTO 500
300 IF S=2IF X=15IF C=5IF Y>-5IF Y<5GOTO 500
310 IF S=3IF X=15IF C=5IF Y=0IF B=0GOTO 500
320 BOX U,V,3,3,3;BOX M,N,8,2,3;BOX M+2,N+1,1,2,3
330 IF I#0BOX U-Ib2,V,3,1,3
340 IF J#0BOX U,V-Jb2,1,3,3
350 M=M+D;N=N+E
360 GOTO 200
500 CLEAR ;&(21)=0;O=O+1
510 CY=0;PRINT "IN ORBIT NEAR THE CYGUS"
520 FOR A=1TO 500;NEXT A;CLEAR
530 FOR A=1TO 50;BOX RND (160)-80,RND (80)-40,1,1,1;NEXT A
540 BOX 0,0,100,10,1
550 BOX 50,7,40,10,1
555 BOX 0,30,15,15,1;BOX 0,30,11,11,2
560 BOX 50,-7,40,10,1
570 BOX 0,10,140,1,1
580 BOX 0,-10,140,1,1
590 BOX 40,15,8,10,1
600 LINE -70,10,0
610 FOR A=-65TO 65STEP 10
620 LINE A,-10,3;LINE A+5,10,3
640 NEXT A
650 LINE -70,-10,0
660 FOR A=-65TO 65STEP 10
670 LINE A,10,3;LINE A+5,-10,3
690 NEXT A
698 X=-70;Y=-30
700 FOR A=1TO 250
710 BOX X,Y,3,3,3
720 BOX X,Y,3,3,3
730 X=X+JX(1);Y=Y+JY(1)
740 NEXT A
900 FOR A=1TO 2000;NEXT A
910 GOTO 100
1000 CLEAR ;H=H+1;&(21)=0
1010 PRINT "  LOST IN THE BLACK HOLE"
1015 FOR A=1TO 500;NEXT A;CLEAR
1020 BOX 0,10,30,20,1;BOX -20,-30,10,6,1;BOX 20,-30,10,6,1
1021 LINE -10,0,1;LINE -20,-30,1;LINE 20,-30,1;LINE 10,0,1
1030 FOR A=200TO 0STEP -1
1040 BC=A;&(9)=192;&(0)=A+18;&(1)=A+23
1050 &(2)=A+35;&(3)=A+J8;&(9)=40
1060 &(20)=A;&(18)=A;&(16)=A
1070 NEXT A;&(22)=0;CLEAR ;GOTO 900
```

```
 1 .
 2 .
 3 .SLOT
 10 CLEAR
 20 PRINT ;PRINT
 30 PRINT "     SLOT MACHINE"
 40 PRINT "  1.   PULLS LEVER"
 45 PRINT "  TRIGGER PULLS LEVER"
 50 FOR A=1TO 1500
 60 NEXT A
 95 CLEAR ;P=1000;W=0;L=0
100 BOX 10,20,100,40,1
110 FOR A=40TO 0STEP -5
120 BOX 55,A,5,5,2
130 NEXT A
140 FOR A=0TO 40STEP 5
150 BOX 55,A,5,5,1
160 NEXT A
170 CX=-10
180 CY=-10
190 PRINT #1,P
200 FOR A=0TO 10
210 NT=A
220 B=RND (15)+88;C=RND (15)+88;D=RND (15)+88
230 CX=-20
231 CY=20
232 TV=B
233 CX=0
234 CY=20
235 TV=C
236 CX=20
237 CY=20
238 TV=D
240 NEXT A
245 NT=3
250 IF B=CIF C=DGOTO 1000
260 IF B=CGOTO 1100
270 IF B=DGOTO 1100
280 IF C=DGOTO 1100
290 IF B=100GOTO 1200
300 IF B=101GOTO 1200
310 IF B=102GOTO 1200
320 IF B=103GOTO 1200
330 CX=-30
332 CY=10
333 PRINT #1,"YOU LOSE"
334 L=L+1
335 CX=50
336 CY=-30
337 PRINT #1,L
340 FOR A=1TO RND (10)+5
350 P=P-RND (10)
360 CX=-10
370 CY=-10
380 PRINT #1,P
390 NEXT A
400 IF P<1GOTO 2000
```

```
800 CX=-70
801 CY=-30
802 PRINT #1,W
810 IF &(23)=8GOTO 100
812 IF P>30000GOTO 900
815 IF &(23)=4GOTO 330
820 IF &(23)=2GOTO 1000
830 IF &(22)=2GOTO 1100
840 IF &(21)=2GOTO 1200
845 IF TR(1)GOTO 100
850 GOTO 810
900 FOR A=1TO 50
910 PRINT " BUSTED ",
920 NEXT A
930 GOTO 2020
1000 CX=-30
1001 CY=10
1010 PRINT #1,"JACKPOT"
1020 FOR A=1TO RND (50)+50
1030 FC=RND (32)b8+2;BC=RND (32)b8-1
1040 P=P+RND (500)+50
```

```
1050 CX=-10
1060 CY=-10
1070 PRINT #1,P
1075 IF P>30000GOTO 900
1080 NEXT A
1085 W=W+1
1090 GOTO 800
1100 CX=-30
1101 CY=10
1102 PRINT #1,"WINNER"
1109 FOR A=1TO RND (10)+5
1110 P=P+10
1120 CX=-10
1130 CY=-10
1135 FC=RND (32)b8+2
1140 PRINT #1,P
1150 NEXT A
1160 W=W+1
```

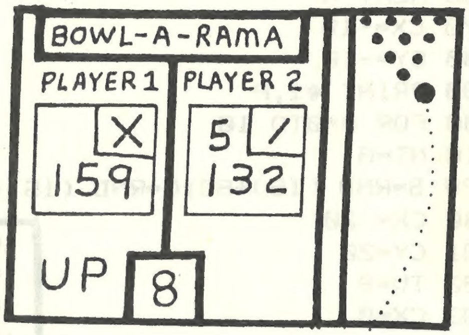
```
1170 GOTO 800
1200 CX=-30
1201 CY=10
1202 PRINT #1,"WIN"
1210 FOR A=1TO RND (10)+5
1220 P=P+RND (10)
1230 CX=-10
1235 CY=-10
1240 PRINT #1,P
1250 NEXT A
1260 W=W+1
1270 GOTO 800
2000 CLEAR
2010 PRINT "YOU BLEW IT"
2020 PRINT "    TURKEY"
```

BOWL-A-RAMA (p.52)

By Bob Hensel

BOWL-A-RAMA is a two player game. The computer displays the pins and keeps score. The ball is invisible at the bottom of the screen moving between the two gutters. When the player UP pulls his trigger the ball appears and starts rolling down the alley. The player controls the curve on the ball by moving his joy stick left or right. The frame number is shown in the center box at the bottom of the screen.

```
  1 .BOWL-A-RAMA
  2 .BY BOB HENSEL
  3 :RETURN
  5 &(9)=28;&(0)=25;&(1)=25;&(2)=31;&(3)=31;BC=6;FC=0;CLEAR
 10 B=0;C=0;D=0;F=0;G=0;H=0;I=0;P=5;T=0;Q=0;R=0
 20 BOX -28,38,86,12,3;CX=-64;CY=38;PRINT "BOWL-A-RAMA";CX=-77;CY=24;PRINT "PLA
YER 1  PLAYER 2"
 40 BOX -54,0,35,35,3;BOX -54,0,31,31,3;LINE -54,16,4;LINE -54,5,1;LINE -38,5,1
;BOX 0,0,35,35,3;BOX 0,0,31,31,3;LINE 0,16,4;LINE 0,5,1
 50 LINE 16,5,1;BOX -27,-30,30,16,3;BOX -27,-30,26,12,3;BOX -27,5,2,54,3
 310 BOX 55,0,34,87,1;BOX 41,0,1,87,3;BOX 68,0,1,87,3;E=2;GOSUB 1000
 320 E=1
 440 CX=R-57;CY=-33;PRINT "  ";IF P>3P=0;F=F+1;Q=1;CX=-28;CY=-31;PRINT #1,F;R=0
 450 P=P+1;IF P>2Q=3;R=54
 455 IF P-Q=0CX=R-65;CY=10;PRINT " ";CX=R-47;CY=10;PRINT " "
 460 CX=R-57;CY=-33;PRINT "UP"
 500 IF Q=1V=1;GOTO 550
 510 V=2
 550 A=RND (23)+43;IF TR(V)=0GOTO 550
 620 A=A+JX(V)
 630 BOX A,B-37,2,2,2;B=B+8;IF B>70GOTO 800
 650 IF A<44A=40;MU="1";GOTO 630
 660 IF A>66A=70;MU="1";GOTO 630
 670 GOTO 620
 800 BOX 55,-7,26,72,1;IF A<45GOTO 1100
 802 IF T=8IF A<64IF A>47GOTO 1100
 810 IF A<48GOTO 1020
 820 IF A<51GOTO 1010
 830 IF A<54GOTO 1000
 835 IF A<55GOTO 1070
 840 IF A<57IF P-Q=0E=2;GOTO 1070
 843 IF A<57GOTO 1070
 845 IF A<58GOTO 1070
 850 IF A<61GOTO 1030
 860 IF A<64GOTO 1040
 870 IF A<67GOTO 1050
 880 GOTO 1100
 900 Y=30;X=55;W=0
 910 IF PX(X,Y)=0W=W+1
 920 X=X+2;IF X>64GOTO 940
 930 GOTO 910
 940 X=46;Y=Y+4;IF Y<43GOTO 910
 950 IF P-Q=0W=10-W;GOTO 970
 955 IF W=0T=10-T;GOTO 980
 960 W=10-T-W
 970 T=W
 980 RETURN
1000 BOX 52,34,2,2,E;BOX 55,38,2,2,E;BOX 58,42,2,2,E
1010 BOX 49,38,2,2,E;BOX 52,42,2,2,E
1020 BOX 46,42,2,2,E;IF E#2GOTO 1100
1030 BOX 58,34,2,2,E;BOX 55,38,2,2,E;BOX 52,42,2,2,E
1040 BOX 61,38,2,2,E;BOX 58,42,2,2,E
1050 BOX 64,42,2,2,E;IF E#2GOTO 1100
1060 BOX 55,30,2,2,E;RETURN
1070 BOX 55,35,22,16,1;IF E=1GOTO 1100
1080 BOX 46,42,2,2,E;BOX 64,42,2,2,E;E=1;GOTO 1100
1100 GOSUB 900
1105 IF Q=1J=C;K=D;GOTO 1120
1110 J=G;K=H
```



```
1120 IF F#11J=J+T
1130 IF K>0J=J+T;K=K-1;IF K>1GOTO 1130
1140 CX=R-60;CY=-5;PRINT #1,J;IF P-Q=1GOTO 1200
1170 IF T#10CX=R-65;CY=10;PRINT #1,T;GOTO 1300
1180 CX=R-47;CY=10;PRINT #1,"X";IF F=11GOTO 1300
1190 K=K+2;P=P+1;GOTO 1300
1200 CX=R-47;CY=10;IF W#0PRINT #1,T;GOTO 1300
1205 IF T=0PRINT #1,T;GOTO 1300
1210 PRINT #1,"/";IF F#11K=K+1
1300 IF Q=1C=J;D=K;GOTO 2000
1310 G=J;H=K;GOTO 2000
2000 B=0;IF F<10GOTO 3000
2010 IF P-Q=0GOTO 3000
2030 IF K=0GOTO 2070
2035 IF K>1P=P-1
```

[instructions on p. 51]

```
2050 P=P-1;F=11;GOTO 3030
2070 F=10;IF P=4GOTO 5000
3000 IF P-Q=0GOTO 440
3030 T=0;GOTO 310
5000 IF TR(1)=0GOTO 5000
5010 GOTO 5
>
```

```
  1 .
  2 . MASTERMIND
  3 . CAROL BURKEMPER
 10 CLEAR ;T=0;BC=140;FC=0
 15 @(1)=RND (8)+48
 20 FOR Z=2TO 5
 30 @(Z)=RND (8)+48
 35 BC=(BC+1)
 40 FOR Y=1TO Z-1
 50 IF @(Z)=@(Y)GOTO 30
 60 NEXT Y;NEXT Z
 70 FOR R=33TO -39STEP -8
 80 H=0;J=0;S=15;C=-63;L=77
 90 BC=141;T=T+1
 95 BOX 3,R,139,7,2
100 FOR B=11TO 15
110 A=KP;@(B)=A
115 IF A=57GOTO 500
120 CY=R;CX=C;TV=A
130 D=KP;IF D=31GOTO 110
140 IF D#13GOTO 130
150 C=C+14;NEXT B
160 FOR W=1TO 5
170 FOR B=11TO 15
180 IF @(W)=@(B)GOSUB 300
190 NEXT B;NEXT W
200 IF H>0GOSUB 350
210 IF J>0GOSUB 400
220 IF H=5GOTO 250
230 NEXT R
240 GOTO 70
250 BC=241;FC=7
260 PRINT ;PRINT ;CX=-50;PRINT #2,"CORRECT IN",T
265 PRINT ;PRINT "    PRESS 'GO' TO START
270 A=KP;IF A=13GOTO 10
280 GOTO 270
300 IF B<15IF @(B)=@(B+1)RETURN
310 IF @(W)=@(W+10) H=H+1;RETURN
330 J=J+1;RETURN
350 FOR N=1TO H
360 NT=6;MU=L;MU=(L-1);MU=(L+2);L=L-8;NT=3
370 BOX S,R,6,6,1;S=S+15
380 NEXT N;RETURN
400 FOR N=1TO J
410 NT=7;MU=L;MU=L;MU=(L-2);L=L-8;NT=3
420 BOX S,R,6,6,1;BOX S,R,4,4,2;S=S+15
430 NEXT N;RETURN
500 FOR G=1TO 5
510 CY=(R-8);CX=C
520 TV=@(G)
530 C=C+10
540 NEXT G
550 C=-63;GOTO 110
RUN
```

PROJECT ONE is on schedule. We expect to see the printed circuit boards sometime in June.

BIORYTHM of p.44 should have been noted as being the original output of Rich Tietjens. I've had some questions about the required operations to load it and the data, and so here is a clarification...
On page 38, the four line program is a general method for loading data, initially described early last year, into the @ strings so that you can store them. In this case 87 items are stored. Line numbers only create a program, and are not necessary for a one-time job. Confusion arose because the numbers duplicate those in the base program. To load the data for the Biorythm program, use the following scheme - punch in the program from line 10 to 1160. Then punch in the following which will only work once -
:PRINT;LIST; FOR A = Ø TO 86; PRINT #1,"@(", A, ")=",; INPUT @(A); NEXT A; PRINT " CLEAR; RUN" and then stop. Now start the Record function on your tape recorder and then punch GO. The Bally will read the :PRINT; and send data to controller port 3. It will read LIST; and list the program. And when that is done, it will do the A loop. You will see @ (Ø) = @(A) ■ on the screen, meaning that the computer is waiting for you to insert the value of @(Ø) per the table. Page 45 lists this as Ø, so enter Ø and press GO. The machine will then print @ (1) = @(A) ■ , asking for the second value, enter 1Ø and GO, then 19 and GO, etc. to the end. As you are doing this, the data is going into the computer and at the same time into the tape. You can now run the program. When you load the program from the tape into the computer at some future time, you will again see the @ statements, but with the answers as well. And you will probably get a WHAT? along with each one, disregard these.

MEMORY TUTORIAL #3

The types of memory have been covered, now lets discuss the operation. The chip that contains the memory bank is, in itself, passive. That is, it requires an outside stimulus to make it work. It won't do anything on its own. The Central Processing Unit, CPU, is the controlling agent - in our case the Zilog Z-80 This device is factory-instructed to perfrom all the necessary operations, and is the heart of the computer. Programs in memory, ROM or RAM, are stored as a series of Ø's and 1's. The CPU looks at the memory locations in sequence, and reads, interprets, and executes the program, one step at a time. Each Ø and 1 is called a bit and eight of these make a byte. Instructions usually require only one byte, but some complicated ones can take up to four. The Z-80 keeps reading the instructions as it goes along until it is told to STOP, or the program ends, or it is told to go to another memory location by a GOTO or GOSUB instruction. Memory locations contain the following types of bytes: Instructions, Data, Address, Device Code, and Displacement.

The CPU has an INSTRUCTION SET, developed by the company engineers, that is a sort of menu of the things that it can do. These can be obtained from the manufacturer or some data books. This Set is used by the programmer to make up a program in Assembly Language.

The CPU operates only in binary notation - Ø's and 1's, eight in a row, and this is called Machine Language. "Ø111Ø11Ø" and "ØØ11111Ø" are examples. With this scheme, it is difficult for the programmer to avoid errors. To make the operation somewhat simpler, the same information is converted to hexadecimal notation, reducing the 8-unit figure to a 2-unit figure. The examples above convert to "76" and "3E".

Combine the hex notation with a mnemonic (an aid to memory, don't pronounce the initial m) yields Assembly Language. This is still somewhat awkward for the casual operator such as ourselves to use. The above examples become "76 HALT" and "3E LD A,n".

The next level up is a High Level Language such as BASIC, FORTRAN, COBOL, that we use to talk to the computer. A few words in this language require a lot of assembly or machine language commands, to be illustrated below.

PROGRAMMING- The CPU needs ADDRESS and DATA information along with the instructions. Data is material that is generally stored by the user as a result of program input (what is your age? ___), but some computers may have tables or conversion constants stored, etc. Address information tells it where to look in memory for specific items.

Registers are areas within the CPU that are used as temporary storage positions. If you wanted to operate on any memory item, it must be taken from memory and loaded into a register, operated upon, and then stored somewhere back in memory. The Z-80 has 12 registers available in two sets. A number 'taken' from memory does not erase that memory slot - the CPU merely reads the number and duplicates it in the register.

To add two numbers together, we would write

```
10 INPUT A,B
20 C = A + B
```

While the computer programmer had to give this much detail to the CPU:

THIS EXAMPLE JUST COVERS THE STEP 10 INPUT A, B

FIRST IN ASSEMBLY LANGUAGE :-

```
STEP |←— ASSEMBLY ———→|

0000  3E  05        LD A, 05D       Load Register A with decimal 5 in hex format
0002  32  00  01    LD(0100), A     Load Memory location 0100 with contents of Register A
0005  3E  0A        LD A, 10D       Load Register A with decimal 10 in hex format
0007  32  10  01    LD (0110), A    Load Memory location 0110 with contents of Register A
```

IN BINARY, THE ABOVE GETS WORSE:

```
STEP    BINARY

0000    00111110    Load Register A with
0001    00000101    decimal 5 in binary format
0002    00110010    Load Memory Location        → with contents of Register A
0003    0000000                           ↓ ↗00
0004    0000001                           01
0005    00111110    Load Register A with
0006    00001010    decimal 10 in binary format
0007    00110010    Load Memory Location        → with contents of Register A
0008    00001010                               10
0009    0000001                           ↓ ↗
                                          01
```

ALL THIS TO LOAD TWO NUMBERS ——

PROJECT FOUR refers to a mini-memery scheme new underway that will previde an additienal 1K ef RAM and 1K ef ROM. A pretetype has been built and is eperatienal. By increasing memery by 50%, the casual Bally user will be able te enjey greater detail ef pregram cemplexity, depth, and features at a small cest. We are new casting abeut fer parts in quantity, and expect te have a cemplete kit available fer abeut 75. by mid-summer. This unit plugs inte the 50-pin cennecter in the back, and has a pig tail te the Light Pen cennecter fer pewer.

WIRING Is anyene interested in centract werk te wire up seme ef these kit prejects? Repetitieus werk, but must be ef highest caliber fer reliability purpeses.

CREDIT Almest all the pregrams submitted te me are the effert ef ene er anether ef eur subscribers. Once in a while I receive a pregram that has been cenverted frem ene published in a magazine er ether seurce. In erder te previde preper credit te the eriginater, please indicate whether the pregram yeu are submitting is eriginal er a cenversien, variatien, medificatien.,etc.

ADS V. Jupe reports that he no longer has any used Bally units.

New Programs! Bowling, Keno and more. Send for complete descriptions
Scott Waldinger 24740 Woodcroft Dr. Dearborn MI 48124

Bally is unfixably broken, so I am selling my carts, BASIC, Interface, and software at low prices. In good condition. For more info write
Joel Jacobs 9727 Blantyre Dr. Beverly Hills, CA 90210

LIFE IV and data on cassette or open reel w/documentation on program and game $3.50 your tape, $5.50 on mine Matt Giwer, 3922 Millcreek Dr. Annandale, VA 22003

56-key Clare-Pender ASCII keyboard (#KB-6) - new surplus. Upper Case with instructiens te add lewer case with ene switch. Ne Case. One only. 60.
W. Munn N111-W15831 Vienna Ceurt #7 Germantewn WI 53022

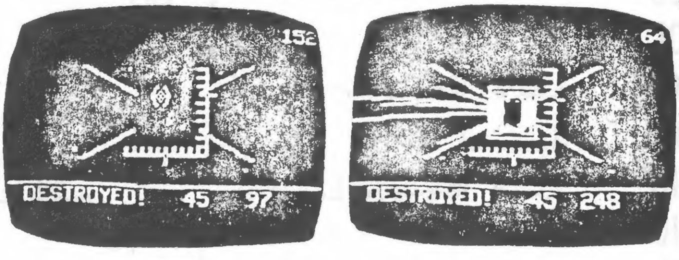
FIRST ef 4 program packages new available. Each package will centain 2, 4, er mere programs, ene per side ef tape, instructiens, documentatien, etc. 9.95 each - Cest Effective Cemputer Services 1041 Ute, Grand Junctien CO 81501

NEW PROGRAM!... Code: BASIC/MACHINE LANGUAGE

STAR FIRE!-- 1 player. This program is the most unique of them all. It uses a user-defined subroutine in MACHINE LANGUAGE to generate the super fast-acting characters which appear on the screen for you to shoot down or be shot down by!! The game consists of 3 characters which appear in random locations and continually travel in a chosen direction, an enemy fighter, enemy refueling station, and a meteor/planet. The characters change in size (up to 4X) according to the distance you are from them. You move your hand-controller in the direction you would if you were in a plane, line him up on the gunsight and fire at him while you're lineing him up. You can actually fire on him at the same time he is fireing on you, both shots take a few seconds to reach their destination. The background stars MOVE in relation with your ship. Background color changes with the point values. The faster you hit him the more points you get. The graphics are spectacular There are 2 explosion subroutines (visual & sound). The Machine Language program uses the input buffer for storage, but is in the back end of the buffer so you can still use the keypad. Listings of both the Basic & Machine language are given, along with 2 pages of program description, variable index & other specs. This is a multi-processing program & should have great educational value in the 13 subroutines alone. The programs use ALL free memory for Basic storage and most of the buffer, & all of the scratchpad. This may be your most often played game, you can study it for hours. It took 4 MONTHS to develop, so you can be sure that the techniques are far advanced and everything runs glitch-free. Those who want to know how to do copy-protection, this program has it! ONLY- $6.50

STAR FIRE ASSEMBLER- The program that wrote the machine language program is available for- $2.50

NEW Game
STAR FIRE Program with
MACHINE LANGUAGE $6.50



From:
SEBREE'S COMPUTING
456 Granite Ave.
Monrovia, CA 91016

ARCADIAN™

Robert Fabris, taxed
3626 Merrie Dr.
San Jose, CA, 95127

FIRST CLASS